PERGAMON

Computers & Industrial Engineering 39 (2001) 187–211

computers & industrial engineering

www.elsevier.com/locate/dsw

Comparison of constraint logic programming and distributed problem solving: a case study for interactive, efficient and practicable job-shop scheduling

D. Trentesaux*, P. Pesin, C. Tahon

Equipe Système de Production, Laboratoire d'Automatique et de Mécanique Industrielles et Humaines, UMR CNRS 8530, Le Mont Houy, F-59313 Valenciennes cedex 09, France

Abstract

The job-shop scheduling issue is more and more described not only in terms of efficiency (e.g. Makespan), but also in terms of interactivity and practicability. The aim of this paper is to evaluate the ability of two approaches inherited from artificial intelligence domain to contribute to the solving of this issue: Constraint Logic Programming (CLP) on the one hand, and, on the other hand, Distributed Problem Solving (DPS) inherited from Distributed Artificial Intelligence (DAI). This analysis is achieved through the use of two specific computerised tools: Constraint Handling In Prolog (CHIP) as a CLP software and the Distributed Production Scheduling System (DPSS) as a distributed problem solving system. These tools are then evaluated in terms of interactivity, efficiency and practicability. Interactivity is discussed according to qualitative points of view such as the ability to provide efficient decision support, a set of alternative solutions and the possibility to parameterise the algorithms. Efficiency is described in terms of optimality or sub-optimality by the analysis of the Makespan criterion vs. fixed computation time. Practicability is associated to the industrial viability of the methods: ability to cope with real industrial case study or ability to face real industrial contexts. Evaluation is then performed through a multiple criteria analysis. This analysis is achieved given an increasing number of operations to perform.

The results highlight the high complementary level of these two approaches, allowing us to provide a framework for a joint integration, which shall be optimised when taking into account the assets of each approach according to the three evaluation criteria.

Keywords: Constraint logic programming; Distributed problem solving; Job-shop; Scheduling

1. Introduction

The domain addressed in this paper is the scheduling of job-shop production systems. A production

* Corresponding author. Fax: +33-3-2751-1310.

E-mail addresses: damien.trentesaux@univ-valenciennes.fr (D. Trentesaux), philippe.pesin@univ-valenciennes.fr (P. Pesin), christian.tahon@univ-valenciennes.fr (C. Tahon).

0360-8352/01/$ - see front matter
PII: S0360-8352(00)00078-4

system can be defined as a system composed of technical and human resources responsible for the physical realisation of products satisfying customers from raw materials or semi-finished products. Each product is described by a manufacturing order, composed of tasks named "operations". A job-shop production system is characterised by the existence of operations that can be clearly identified and separated, in a discontinuous sense, such as manufacturing systems. Architectures of job-shops are generally more elaborated than continuous production systems (e.g. resource organisation, routing).

A production management system is required to elaborate production schedules, to control and to evaluate production progress since:

- products must satisfy customers' needs in terms of quality, functionality, cost and delay;
- production systems are characterised by constraints (production times, costs, etc.), limited resources capabilities and suffers from disturbances.

Thus we disassociate the production system, responsible for the production progress, from the management system, responsible for the management of the production progress. In this paper, we only focus on the scheduling of the production, that is, we do not deal with design phase nor production activity phase.

It is nowadays clear that the need of "intelligent" production management structures continuously increases because of the decrease of product life cycle and the increase of competition. As a consequence, a management system is now evaluated not only in terms of completion time efficiency (product time to market, etc.), inventory level, but also in terms of interactivity in order to integrate all the wishes of the managers (as well as structured or ill- structured wishes) and industrial viability to face the largest number of real production system constraints, whatever their specificity could be.

The scope of this paper is then to evaluate two approaches inherited from artificial intelligence, Constraint Logic Programming (CLP) and Distributed Problem Solving (DPS) according to three criteria (axis of significance) listed below:

- Scheduling efficiency: we focus in this paper on the Makespan criterion (Cmax), which can be in a first step considered as representative of the efficiency of the scheduling (optimality proved or not).
- Interactivity: it has been proved that integration of the human operator in the management system is a way to provide a realistic scheduling elaboration of a job-shop production system (Trentesaux, Moray & Tahon, 1998), implying a need of interaction capability. In this paper, measurement of interactivity refers to the following axis of significance: average time used by the algorithm, capability to modify the parameterisation of the method, tools for decision support, possibility to integrate ill-structured data (lack of knowledge, uncertainty, etc.), existence of any graphic capabilities or ergonomic considerations, level of cooperation performed, etc.
- Practicability and industrial viability: that is, the ability of the approach to manage real industrial case studies, to face real industrial contexts and constraints and to manage real sets of data and parameters.

The first part deals with the distributed problem solving approach and the Distributed Production Scheduling System (DPSS) tool used while the second one deals with the CLP approach and the Constraint Handling In Prolog (CHIP) tool.

The third part details the description of the problem and the evaluation method. We analyse the